BEEP
Fonction: Délivre un son par le haut-parleur
Format: BEEP <hauteur de son>, <durée>
Exemple: BEEP 100,2

CIRCLE
Fonction: Trace un cercle
Format: CIRCLE [STEP] (<coordonnée x>, <coordonnée y>), <rayon>
Exemple: CIRCLE (98,16),8

CLEAR
Fonction: Initialise les variables et définit la zone de mémoire à utiliser.
Format: CLEAR [<variable de caractères> [,<limite supérieure de la zone réservée à l'utilisateur>]]
Exemple: CLEAR 100,500

CLOAD
Fonction: Charge un fichier à partir d'une bande cassette
Format: CLOAD ["<désignation de fichier>"]
Exemple: CLOAD "PROG-1"

CLOAD?
Fonction: Vérifie la présence d'un fichier sur bande cassette
Format: CLOAD? ["<désignation de fichier>"]
Exemple: CLOAD? "PROG-1"

CLS
Fonction: Affiche l'écran.
Format: CLS
Exemple: CLS

CONSOLE
Fonction: Fixe l'état de la console
Format: CONSOLE [<ligne de début du roulement>] [,[<nombre de lignes de roulement>] [,[<commutateur d'affichage des touches>] [,[<commutateur de déclic des touches>] [,[<commutateur de répétition des touches>]]]]]
Exemple: CONSOLE 1,2,1,0,1

CONSOLE @
Fonction: Met en et hors fonction le haut-parleur, initialise les touches définies par l'utilisateur et fixe le mode du clavier
Format: CONSOLE @ [[<commutateur de réveil>] [,[<définition>] [,[<mode du clavier>]]]]
Exemple: CONSOLE @,,3

CONT
Fonction: Reprend l'exécution du programme après son arrêt.
Format: CONT
Exemple: CONT

CSAVE
Fonction: Sauvegarde le programme en zone texte de la mémoire sur bande cassette.
Format: CSAVE "<désignation de fichier>"
Exemple: CASVE "PROG-1"

DATA
Fonction: Définit les valeurs à lire par l'instruction READ.
Format: DATA <constante> [,<constante>...]
Exemple: DATA 1, "X-07", "CANON COMPUTER",3

DEFFN
Fonction: Définit une fonction.
Format: DEFFN <nom> [(<argument> [,<argument>...])] = <définition de la fonction>
Exemple: DEFFNZ (X, Y) =X*2+Y*3+6

DEFINT/SNG/DBL/STR
Fonction: Déclarent les types de variable.
Format: DEF <type> <gamme(s) de lettres>
Exemple: DEFINT I-N,X

DELETE
Fonction: Supprime un fichier de la zone pour fichiers en RAM.
Format: DELETE "<désignation de fichier>" ["<type de fichier>"]
Exemple: DELETE "SAMPLE", "P"

DIM
Fonction: Déclare un ou plusieurs tableaux.
Format: DIM <nom de variable> (<valeur maximum de l'indice> [,<valeur maximum de l'indice>...]) [,<nom de variable> (<valeur maximum de l'indice> [,<valeur maximum de l'indice>...])...
Exemple: DIM A$(100), B(18,16)

DIR
Fonction: Affiche le contenu de la zone pour fichiers en RAM
Format: DIR [# <numéro de fichier>]
Exemple: DIR #1

END
Fonction: Termine l'exécution d'un programme.
Format: END
Exemple: END

ERASE
Fonction: Elimine un ou plusieurs tableaux.
Format: ERASE <nom de variable en tableau> [,<nom de variable en tableau>...]
Exemple: ERASE A, B$

ERROR
Fonction: Simule l'occurence d'une erreur.
Format: ERROR <numéro d'erreur>
Exemple: ERROR 200

EXEC
Fonction: Exécute un programme en langage machine.
Format: EXEC <adresse de départ>
Exemple: EXEC &H3000

FOR TO STEP...NEXT
Fonction: Exécute une série d'instructions de manière répétitive.
Format: FOR <variable> = <valeur initiale> TO <valeur finale> [STEP<incrément>]
⋮
NEXT
Exemple: FOR I = 1 TO 10: NEXT

FSET
Fonction: Fixe la dimension de la zone pour fichiers en RAM
Format: FSET <dimension de la zone pour fichiers>
Exemple: FSET 1024

GOSUB ~ RETURN

Fonction: Exécute un sous-programme.
Format: GOSUB <numéro de ligne>
Exemple: GOSUB 1000

GOTO

Fonction: Change le déroulement du programme.
Format: GOTO <numéro de ligne>
Exemple: GOTO 2000

IF ~ THEN ~ ELSE

Fonction: Décide de la condition spécifiée par une expression logique
Format: IF <expression logique>
|THEN |<instruction> | |[ELSE |<instruction> |]
| |<numéro de ligne>| | |<numéro de ligne>|
|GOTO <numéro de ligne> | |
Exemple: IF A=0 THEN 100 ELSE A=0:GOTO 200

INIT

Fonction: Met un dispositif en service.
Format: INIT # <numéro de fichier>, "<descripteur de fichier>"
[,|<dimension de fichier> |[,|"<type de fichier>"|]]
|<cadence de transmission>| |"<mode>" |
Exemple: INIT #2 "FICHIER1", 100,"D"

INPUT

Fonction: Introduit des données dans des variables.
Format: INPUT ["<chaîne d'appel>";] <variable> [,<variable>...]
Exemple: INPUT "VOTRE NOM"; N$

INPUT

Fonction: Lit des données dans un fichier et les attribue à des variables.
Format: INPUT # <numéro de fichier>, <variable> [,<variable>...]
Exemple: INPUT #1,A,B$

LET

Fonction: Attribue une valeur à une variable.
Format: [LET] <variable> = <expression>
Exemple: LET PI = 3.1415926535898

LINE

Fonction: Trace un trait.
Format: LINE [STEP] [(<coordonnée X>, <coordonnée Y>)]-[STEP](<coordonnée X>, <coordonnée Y>)
Exemple: LINE (0,0)-(119,31)

LINE INPUT

Fonction: Entre un trait entier.
Format: LINE INPUT ["<chaîne d'appel>";] <variable de caractères>
Exemple: LINE INPUT "QUEL EST VOTRE NOM"; N$

LINE INPUT

Fonction: Entre un trait complet à partir d'un fichier.
Format: LINE INPUT # <numéro de fichier> <variable de caractères>
Exemple: LINE INPUT #1, N$

LIST

Fonction: Affiche une liste (complète ou partielle) du contenu de programme en zone texte de la mémoire.
Format: LIST [<numéro de la ligne de départ>] [-[<numéro de ligne finale>]]
Exemple: LIST100-200

LIST @

Fonction: Affiche, ligne par ligne, le contenu de programme en zone texte de la mémoire.
Format: LIST @[<numéro de ligne de départ>] [-[<numéro de ligne finale>]]
Exemple: LIST @ 100-200

LIST

Fonction: Délivre (complètement ou partiellement) le contenu d'un programme en zone texte de la mémoire à la zone pour fichiers.
Format: LIST [@]# <numéro de fichier>, [<numéro de ligne de départ>] [-[<numéro de ligne finale>]]
Exemple: LIST #1,100-

LLIST

Fonction: Délivre (complètement ou partiellement) le contenu d'un programme en zone texte de la mémoire vers une imprimante.
Format: LLIST [<numéro de ligne de départ>] [-[<numéro de ligne finale>]]
Exemple: LLIST

LOAD

Fonction: Charge un programme d'un fichier en zone texte de la mémoire.
Format: LOAD "<descripteur de fichier>" [,
|<dimension de fichier> |
|<cadence de transmission> |
[,|"<type de fichier>"|]]
|"<mode>" |
Exemple: LOAD "OPT:",2400,"B"

LOAD?

Fonction: Vérifie le contenu d'un fichier de programme.
Format: LOAD? "<descripteur de fichier>"[,
|<dimension de fichier> |
|<cadence de transmission> |
[,|"<type de fichier>"|]]
|"<mode>" |
Exemple: LOAD?"CASI:"

LOCATE

Fonction: Déplace le curseur
Format: LOCATE <position horizontale>, <position verticale>
Exemple: LOCATE 10,0

LPRINT

Fonction: Délivre une donnée à l'imprimante graphique.
Format: LPRINT [([<format de caractère>] [,<couleur>])] [,]
[USING "<format de caractère>";] [<expression>
[,<expression>...]]
Exemple: LPRINT [N,0] "CANON";
LPRINT [,2], USING" # # # ";A, B
LPRINT U$

MOTOR

Fonction: Enclenche le moteur d'un enregistreur à cassette.
Format: MOTOR [|ON |]
|OFF|
Exemple: MOTOR ON

NEW

Fonction: Efface le programme en zone texte de la mémoire.
Format: NEW
Exemple: NEW

NEXT

Fonction: Indique la fin d'une boucle.
Format: NEXT [<variable> [,<variable>...]]
Exemple: NEXT I, J

OFF

Fonction: Coupe l'alimentation du X-07
Format: OFF [|1|2|]
Exemple: OFF

ON ERROR GOTO

Fonction: Définit la ligne de destination d'un branchement en cas d'erreur
Format: ON ERROR GOTO <numéro de ligne>
Exemple: ON ERROR GOTO 100

ON ~ GOSUB/ON ~ GOTO

Fonction: Effectue un branchement à une ou plusieurs lignes en fonction de la valeur d'une expression.
Format: ON <expression> |GOSUB|GOTO| <numéro de ligne$_1$> [,<numéro de ligne $_2$>...]
Exemple: ON I GOSUB 100,130,230,250

OUT

Fonction: Délivre une donnée au port de sortie.
Format: OUT <adresse du port>, <expression>
Exemple: OUT &H7F, &H84

OUT

Fonction: Délivre une valeur vers un fichier externe
Format: OUT # <numéro de fichier>, <expression>
Exemple: OUT #1,18

POKE

Fonction: Inscrit une valeur numérique en mémoire.
Format: POKE <adresse>, <expression>
Exemple: POKE &H0C00, &HC9

PRESET

Fonction: Efface un point
Format: PRESET [STEP] (<coordonnée x>, <coordonnée y>)
Exemple: PRESET (40,25)

PRINT

Fonction: Délivre une donnée sur l'écran.
Format: PRINT [<expression> [|,|;| <expression >...]]
Exemple: PRINT "CA"; "NO", "N",3*2
? "CANON"

PRINT USING

Fonction: Délivre des constantes numériques ou de caractères selon un format déterminé.
Format: PRINT USING "<chaîne de format>"; [,<expression>...]
Exemple: PRINT USING "# # # ";A

PRINT

Fonction: Délivre une donnée à un fichier
Format: PRINT # <numéro de fichier>, [USING "<chaîne de format>";] [<expression> [|;|,| <expression>...]]
Exemple: PRINT #1, "CANON"

PSET

Fonction: Trace un point
Format: PSET [STEP] (<coordonnée x>, <coordonnée y>)
Exemple: PSET (40,25)

READ

Fonction: Lit des constantes dans une instruction DATA et les attribue à des variables.
Format: READ <variable> [, <variable>...]
Exemple: READ A, I, M$

REM

Fonction: Insère une remarque.
Format: REM [<remarque>]
Exemple: REM::**COMP**::
'CANON

RESTORE

Fonction: Réutilise un jeu de données.
Format: RESTORE [<numéro de ligne>]
Exemple: RESTORE 1000

RESUME

Fonction: Reprend l'exécution d'un programme après une procédure de traitement d'erreur.
Format: RESUME [|NEXT|<numéro de ligno>|]
Exemple: RESUME 100

RETURN

Fonction: Indique la fin d'un sous-programme.
Format: RETURN
Exemple: RETURN

RUN

Fonction: Exécute le programme dans la zone texte de la mémoire.
Format: RUN [<numéro de ligne>]
Exemple: RUN 1000

RUN "descripteur de fichier"

Fonction: RUN "[RAM:] <désignation de fichier>" ["<type de fichier>"]
Exemple: RUN "RAM:FICHIER1"

SAVE

Fonction: Sauvegarde le programme se trouvant en zone texte de la mémoire sur un fichier externe.
Format: SAVE "<descripteur de fichier>" [,
|<dimension de fichier>|<cadence de transmission>|
[,|"<type de fichier>"|"<mode>"|]]
Exemple: SAVE "OPT:", 1200"B"

SLEEP

Fonction: Conserve l'affichage, les variables, etc. lorsque l'alimentation du X-07 est coupée.
Format: SLEEP
Exemple: SLEEP

STOP

Fonction: Termine l'exécution d'un programme.
Format: STOP
Exemple: STOP

TROFF
Fonction: Termine le mode de mise au point ligne par ligne.
Format: TROFF
Exemple: TROFF

TRON
Fonction: Affiche la ligne étant exécutée.
Format: TRON [# <numéro de fichier>]
Exemple: TRON # 1

Fonctions

ABS
Fonction: Produit la valeur absolue d'un expression numérique.
Format: ABS (<expression numérique>)
Exemple: PRINT ABS (– 1.8)

ALM$
Fonction: Règle l'alarme
Format: ALM$ = "[<année>], [<mois>], [<our>], [<jour de la semaine>], [<heures>], [<minutes>]"
Exemple: ALM$ = "1983//, &H13,8:15"

ASC
Fonction: Produit le code du premier caractère d'une expression en chaîne.
Format: ASC (<expression en chaîne>)
Exemple: PRINT ASC ("A")

ATN
Fonction: Produit la valeur de l'arctangente d'une expression numérique en radians
Format: ATN (<expression numérique>)
Exemple: A = ATN (0.5)

CDBL
Fonction: Convertit la valeur d'une <expression numérique> en une constante de type nombre réel à double précision.
Format: CDBL (<expression numérique>)
Exemple: A # =CDBL (B!)

CHR$
Fonction: Produit le caractère spécifié par son code.
Format: CHR$ (<code de caractère>)
Exemple: PRINT CHR$(&H41)

CINT
Fonction: Convertit la valeur d'une expression numérique en une constante de type entier
Format: CINT (<expression numérique>)
Exemple: A% = CINT (B #)

COS
Fonction: Produit la valeur du cosinus d'une expression numérique en radians
Format: COS (<expression numérique>)
Exemple: A = COS (3.1415926535898/3)

CSNG
Fonction: Convertit la valeur d'une exression numérique en une constante de type nombre entier à simple précision.
Format: CSNG (<expression numérique>)
Exemple: A! = CSNG (B #)

CSRLIN
Fonction: Produit la position verticale du curseur sur l'écran.
Format: CSRLIN
Exemple: Y = CRSLIN

DATE$
Fonction: Règle le calendrier
Format: DATE$ = "[<année>]/[<mois>]/[<jour>]"
Exemple: DATE$ = "1983/8/22"

ERL
Fonction: Produit le numéro de ligne du programme sur laquelle une erreur s'est produite.
Format: ERL
Exemple: L = ERL

ERR
Fonction: Produit le code d'erreur de la cause d'une erreur.
Format: ERR
Exemple: C = ERR

EXP
Fonction: Produit l'exposant de la valeur d'une expression numérique
Format: EXP (<expression numérique>)
Exemple: E = EXP (1)

FIX
Fonction: Produit la partie entière d'une expression numérique
Format: FIX (<expression numérique>)
Exemple: **A = FIX (B)**

FONT$
Fonction: Spécifie un caractère défini par l'utilisateur.
Format: FONT$ (<code de caractère>) = "<variable$_1$>,..., <variable$_8$>"
Exemple: FONT$ (128) = "56,56,16,124,144,40,68,132"

FRE
Fonction: Donne la zone inutilisée de la mémoire en nombre d'octets.
Format: FRE (<expression>)
Exemple: PRINT FRE (A$)

HEX$
Fonction: Produit une chaîne représentant la valeur hexadécimale d'une expression numérique.
Format: HEX$ (<expression numérique>)
Exemple: PRINT HEX$ (255)

INKEY$
Fonction: Produit le caractère de la touche frappée
Format: INKEY$
Exemple: A$ = INKEY$

INP
Fonction: Produit une donnée d'un fichier ou d'une adresse de port.
Format: INP (# <désignation de fichier>)
INP (<adresse de port>)
Exemple: A = INP (&H84)

INSTR
Fonction: Recherche la présence de la chaîne 2 dans la chaîne 1, et s'il la trouve, produit sa position.
Format: INSTR ([<expression numérique>,] <chaîne 1>, <chaîne 2>)
Exemple: N = INSTR (B$, "N")

INT

Fonction: Produit l'entier le plus grand inférieur à une expression numérique.
Format: INT (<expression numérique>)
Exemple: A = INT (B)

KEY$

Fonction: Produit le contenu d'une touche définie par l'utilisateur.
Format: KEY$ (<numéro de touche>) = "<expression en chaîne>"
Exemple: KEY$ (3) = "RUNRUN" + CHR$(13)

LEFT$

Fonction: Produit une chaîne constituée des caractères extrêmes gauches d'une chaîne.
Format: LEFT$ (<chaîne>, <longueur>)
Exemple: PRINT LEFT$ ("CANON", 3)

LEN

Fonction: Produit la longueur d'une chaîne
Format: LEN (<chaîne>)
Exemple: N = LEN (A$)

LOG

Fonction: Produit le logarithme naturel d'une expression numérique
Format: LOG (<expression>)
Exemple: A = LOG (B)

MID$

Fonction: Produit une chaîne composée du nombre de caractères spécifié d'une chaîne.
Format: MID$ (<chaîne>, <expression 1> [,<expression 2>])
Exemple: A$ = MID$ (B$,3,2)

PEEK

Fonction: Produit le contenu de la mémoire.
Format: PEEK (<addresse>)
Exemple: A = PEEK (&H0)

POINT

Fonction: Vérifie la présence d'un point
Format: POINT [STEP] (<coordonnée x>, <coordonnée y>)
Exemple: A = POINT (110,10)

POS

Fonction: Produit la position horizontale du curseur.
Format: POS (<expression>)
Exemple: X = POS (0)

RIGHT$

Fonction: Produit une chaîne composée du nombre spécifié de caractères extrêmes droits d'une chaîne.
Format: RIGHT$ (<chaîne>, <Longueur>)
Exemple: PRINT RIGHT$ ("CANON", 3)

RND

Fonction: Produit des nombres aléatoires
Format: RND (<expression numérique>)
Exemple: A = RND (1)

SCREEN

Fonction: Produit le code d'un caractère affiché sur l'écran.
Format: SCREEN (<position horizontale>, <position verticale>)
Exemple: PRINT SCREEN (X,Y)

SNG

Fonction: Produit la valeur d'une expression numérique
Format: SGN (<expression numérique>)
Exemple: PRINT SGN(– 1.2)

SIN

Fonction: Produit la valeur du sinus d'une expression numérique en radians.
Format: SIN (<expression numérique>)
Exemple: A = SIN(3 . 1415926535898/3)

SNS

Fonction: Introduit une donnée d'un dispositif d'entrée
Format: SNS (# <numéro de fichier> [,<expression>])
Exemple: A = SNS(# 1, &HFF)

SQR

Fonction: Produit la racine carrée d'une expression numérique
Format: SQR (<expression numérique>)
Exemple: A = SQR(2)

START$

Fonction: Définit un programme comme programme de départ.
Format: START$ = [+]<expression en chaîne>
Exemple: START$ = "RUN" + CHR$(13)

STICK

Fonction: Produit l'état du curseur.
Format: STICK (<expression>)
Exemple: A = STICK(0)

STR$

Fonction: Produit une chaîne représentant une valeur numérique.
Format: STR$ (<expression numérique>)
Exemple: A$ = STR$(123)

STRIG

Fonction: Produit l'état de la barre d'espacement et de la touche définie par l'utilisateur n°6.
Format: STRIG (<expression>)
Exemple: A = STRIG(0)

STRING$

Fonction: Produit une chaîne de longueur spécifiée
Format: STRING$ (<expression numérique$_1$>, | <expression en chaîne> | <expression en chaîne$_2$> |)
Exemple: PRINT STRING$ (5,"A")

TAB

Fonction: Fixe un tabulateur
Format: TAB (<expression numérique>)
Exemple: PRINT TAB(8); "CANON"

TAN

Fonction: Produit la valeur de la tangente d'une expression numérique en radians
Format: TAN (<expression numérique>)
Exemple: A = TAN (3.1415926535898)

TIME$

Fonction: Règle les heures, minutes et secondes de l'horloge.
Format: TIME$ = "[<heures>][:[<minutes>][:[<secondes>]]"
Exemple: TIME$ = "23:53:25"

Fonctions

TKEY

Fonction: Produit l'état d'une touche
Format: TKEY (<chaîne>)
Exemple: A = TKEY ("A")

USR

Fonction: Appelle un sous-programme en langage machine.
Format: USR (<adresse de départ>, <argument>)
Exemple: A = USR (&H0C00, B)

VAL

Fonction: Produit la valeur numérique d'une expression en chaîne.
Format: VAL (<expression en chaîne>)
Exemple: A = VAL ("123")

VARPTR

Fonction: Produit l'adresse d'une variable stockée.
Format: VARPTR (<variable>)
Exemple: A = VARPTR (B)

Instructions graphiques

Instruction	Signification · Format · Fonction
A	INITIALISATION TOTALE La plume est levée du papier puis ramenée au coin gauche. Ce point devient l'origine, et le X07-BASIC retrouve le mode texte.
C	CHANGEMENT DE COULEUR Cn (n = 0...noir, 1...bleu, 2...vert, 3...rouge) Choisit la plume de l'imprimante selon le code de couleur.
D	TRACER x_1, y_1...x_n, y_n A partir de la position actuelle de la plume, des traits sont tracés en reliant successivement les points spécifiées.
F	RETOUR A LA LIGNE Le retour du chariot et l'interligne ne sont exécutés que lorsque rotation alpha (Q) est 0.
H	RETOUR A L'ORIGINE La plume est levée du papier et ramenée à l'origine.
I	INITIALISATION La position actuelle de la plume est prise comme l'origine.
J	TRACE RELATIF J$\triangle x_1$, $\triangle y_1$...$\triangle xn$, $\triangle yx$ A partir de la position actuelle de la plume, des traits sont successivement tracés à partir d'un point relatif (xn, yn)
L	TYPE DE TRAIT Lp (p = 0-15) Spécifie le pas des traits pointillés: 0 = trait plein, 1-15 traits pointillés.
M	DEPLACEMENT M x, y La plume est levée et déplacée à (x, y).
P	IMPRESSION P $c_1c_2c_3...C_n$ (n<256) Imprime des caractères pendant le mode graphique.
Q	ROTATION ALPHA Qn (n = 0 - 3) Changement de l'angle des caractères.
R	DEPLACEMENT RELATIF R $\triangle x$, $\triangle y$ Déplace la plume de ($\triangle x$, $\triangle y$) à partir de la position actuelle dela plume.
S	FIXATION DU FORMAT Sn (n = 0 · 15) Changement du format de caractères.

Tableau des codes de caractères

Forme décimale	Forme hexa-décimale	Caractère	Forme décimale	Forme hexa-décimale	Caractère	Forme décimale	Forme hexa-décimale	Caractère	Forme décimale	Forme hexa-décimale	Caractère
0	00	Voir le tableau des codes de contrôle	32	20		64	40	@	96	60	`
1	01		33	21	!	65	41	A	97	61	a
2	02		34	22	"	66	42	B	98	62	b
3	03		35	23	#	67	43	C	99	63	c
4	04		36	24	$	68	44	D	100	64	d
5	05		37	25	%	69	45	E	101	65	e
6	06		38	26	&	70	46	F	102	66	f
7	07		39	27	'	71	47	G	103	67	g
8	08		40	28	(	72	48	H	104	68	h
9	09		41	29	)	73	49	I	105	69	i
10	0A		42	2A	*	74	4A	J	106	6A	j
11	0B		43	2B	+	75	4B	K	107	6B	k
12	0C		44	2C	,	76	4C	L	108	6C	l
13	0D		45	2D	−	77	4D	M	109	6D	m
14	0E		46	2E	.	78	4E	N	110	6E	n
15	0F		47	2F	/	79	4F	O	111	6F	o
16	10		48	30	0	80	50	P	112	70	p
17	11		49	31	1	81	51	Q	113	71	q
18	12		50	32	2	82	52	R	114	72	r
19	13		51	33	3	83	53	S	115	73	s
20	14		52	34	4	84	54	T	116	74	t
21	15		53	35	5	85	55	U	117	75	u
22	16		54	36	6	86	56	V	118	76	v
23	17		55	37	7	87	57	W	119	77	w
24	18		56	38	8	88	58	X	120	78	x
25	19		57	39	9	89	59	Y	121	79	y
26	1A		58	3A	:	90	5A	Z	122	7A	z
27	1B		59	3B	;	91	5B	[	123	7B	{
28	1C		60	3C	<	92	5C	¥	124	7C	|
29	1D		61	3D	=	93	5D	]	125	7D	}
30	1E		62	3E	>	94	5E	^	126	7E	~
31	1F		63	3F	?	95	5F	_	127	7F	¿

Tableau des dispositifs

Nom du dispositif	Signification	Sortie	Entrée	1er paramètre	2ème paramètre
CON:	Console	○	○	—	—
KBD:	Clavier	○		—	—
COM:	E/S série (RS-232C)	○	○	Cadence de transmission: 100-8000 Valeur de défaut: 4800	Mode ACIA A-H Valeur de défaut: B
OPT:	Coupleur optique	○	○	Cadence de transmission: 100-2400 Valeur de défaut: 1200	Mode ACIA A-H Valeur de défaut: B
GPR:	Imprimante graphique		○	—	—
LPT:	Imprimante de type Centronics		○	—	—
PRT:	Imprimante thermique		○	300 baud, fixe	—
CASI:	Entrée d'enregistreur à cassette	○		1200 baud, fixe	Mode B, fixe
CASO:	Sortie vers enregistreur à cassette		○	1200 baud, fixe	Mode B, fix
RAM:	zone pour fichiers en RAM	○	○	Dimension: nombre d'octets	Type de donnée: A-Z Valeur de défaut:B

Tableau des codes de caractères

Forme décimale	Forme hexa-décimale	Caractère	Forme décimale	Forme hexa-décimale	Caractère	Forme décimale	Forme hexa-décimale	Caractère	Forme décimale	Forme hexa-décimale	Caractère
128	80	♠	160	A0	✓	192	C0	タ	224	E0	ö
129	81	♥	161	A1	。	193	C1	チ	225	E1	ô
130	82	♣	162	A2	「	194	C2	ツ	226	E2	ò
131	83	♦	163	A3	」	195	C3	テ	227	E3	y
132	84	○	164	A4	、	196	C4	ト	228	E4	Ç
133	85	●	165	A5	・	197	C5	ナ	229	E5	ç
134	86	A	166	A6	ヲ	198	C6	ニ	230	E6	Ñ
135	87	Â	167	A7	ァ	199	C7	ヌ	231	E7	ñ
136	88	a	168	A8	ィ	200	C8	ネ	232	E8	Γ
137	89	à	169	A9	ゥ	201	C9	ノ	233	E9	Σ
138	8A	â	170	AA	ェ	202	CA	ハ	234	EA	Π
139	8B	á	171	AB	ォ	203	CB	ヒ	235	EB	Ω
140	8C	å	172	AC	ャ	204	CC	フ	236	EC	α
141	8D	a	173	AD	ュ	205	CD	ヘ	237	ED	β
142	8E	Ì	174	AE	ョ	206	CE	ホ	238	EE	γ
143	8F	í	175	AF	ッ	207	CF	マ	239	EF	∂
144	90	ì	176	B0	ー	208	D0	ミ	240	F0	ε
145	91	î	177	B1	ア	209	D1	ム	241	F1	δ
146	92	í	178	B2	イ	210	D2	メ	242	F2	θ
147	93	Ü	179	B3	ウ	211	D3	モ	243	F3	χ
148	94	ü	180	B4	エ	212	D4	ヤ	244	F4	λ
149	95	ù	181	B5	オ	213	D5	ユ	245	F5	μ
150	96	û	182	B6	カ	214	D6	ヨ	246	F6	ρ
151	97	ú	183	B7	キ	215	D7	ラ	247	F7	π
152	98	É	184	B8	ク	216	D8	リ	248	F8	τ
153	99	ë	185	B9	ケ	217	D9	ル	249	F9	φ
154	9A	è	186	BA	コ	218	DA	レ	250	FA	χ
155	9B	ë	187	BB	サ	219	DB	ロ	251	FB	ω
156	9C	é	188	BC	シ	220	DC	ワ	252	FC	ν
157	9D	Ö	189	BD	ス	221	DD	ン	253	FD	ξ
158	9E	ö	190	BE	セ	222	DE	゛	254	FE	ς
159	9F	ò	191	BF	ソ	223	DF	゜	255	FF	÷

Tableau des codes de commande d'imprimante

Code	Fonction
CHR$ (8)	BS (retour d'un caractère)
CHR$ (10)	LF (interligne)
CHR$ (11)	LU (alignement)
CHR$ (13)	CR (retour chariot)
CHR$ (17)	DC1 (mode texte)
CHR$ (18)	DC2 (mode graphique)

Tableau des codes ACIA

Mode	A	B	C	D	E	F	G	H	Opération
EP	0	0	0	0	1	1	1	1	Parité 1 = par 0 = impair
PEN	0	0	1	1	0	0	1	1	Parité 1 = valide 0 = invalide
CBL	0	1	0	1	0	1	0	1	Longueur de bits 1 = 8 bits 0 = 7 bits

Tableau des codes de contrôle

Code	Fonction	Touche correspondante
0	Chaîne nulle	
1		CTRL + A
2	Retourne au haut du mot précédent	CTRL + B
3	Arrête l'exécution	CTRL + C . BREAK
4		CTRL + D
5	Efface la ligne suivante	CTRL + E
6	Va au haut du mot suivant	CTRL + F
7	Produit un son par le haut-parleur	CTRL + G
8	Efface le caractère précédent	CTRL + H
9	Tabulateur	CTRL + I
10	Interligne	CTRL + J
11	Ramène le curseur à l'origine	CTRL + K . HOME
12	Efface l'écran et ramène le curseur à l'origine.	CTRL + L . CLR
13	Retour du chariot et entrée d'une ligne	CTRL + M . RETURN
14		CTRL + N
15		CTRL + O
16	Efface tout ce qui suit le curseur.	CTRL + P
17		CTRL + Q
18	Introduit un espace.	CTRL + R . INS
19	Arrête temporairement l'exécution d'un ordre ou programme	CTRL + S
20		CTRL + T
21	Efface la ligne où se trouve le curseur, et ramène celui-ci au début de la ligne.	CTRL + U
22	Efface le caractère à l'endroit du curseur.	CTRL + V . DEL
23		CTRL + W
24	Déplace le curseur vers la fin de la ligne.	CTRL + X
25		CTRL + Y
26		CTRL + Z
27		
28	Déplace le curseur d'un rang vers la droite	▶
29	Déplace le curseur d'un rang vers la gauche	◀
30	Déplace le curseur d'une ligne vers le haut	▲
31	Déplace le curseur d'une ligne vers le bas	▼

Tableau des messages d'erreur

11	/0	**Division par zéro** Une division par 0 a été exécutée.
25	BF	**Mode de fichier incorrect** Utilisation d'un fichier de structure incorrecte.
9	BS	**Indice incorrect** Utilisation d'un indice de tableau incorrect.
17	CN	**Poursuite impossible** L'exécution du programme ne peut pas être reprise.
10	DD	**Définition en double** Un tableau ou une fonction définie par l'utilisateur a été défini à deux reprises.
5	FC	**Appel d'une fonction interdite** Une instruction ou une fonction a été appelée incorrectement.
12	ID	**Mode direct interdit** Une instruction est introduite en mode direct alors que cela est interdit.
22	IO	**Erreur de dispositif E/S** Une erreur se produit lors de sortie ou d'entrée d'un dispositif.
27	IR	**Déroulement interdit** Tentative de déroulement direct d'un fichier dont le contenu est incorrect.
L5	LS	**Chaîne trop longue** Utilisation d'une chaîne trop longue.
23	MO	**Opérande manquant** Un paramètre nécessaire a été oublié.
24	NE	**Fichier inexistant** Le fichier demandé n'existe pas.
1	NF	**NEXT sans FOR** L'instruction FOR manque.
26	NO	**Fichier non ouvert** Utilisation d'un numéro de fichier qui n'a pas été défini.
19	NR	**Pas de RESUME** L'instruction RESUME manque dans un sous-programme de traitement d'erreur.
4	OD	**Données manquantes** Les données devant être lues par l'instruction READ ne sont pas fournies.
7	OM	**Mémoire épuisée** Mémoire insuffisante
14	OS	**Place en chaîne épuisée** La place dans la chaîne est insuffisante.
6	OV	**Débordement** Des valeurs numériques sortent de la gamme admise.
3	RG	**RETOUR sans GOSUB** L'instruction RETURN est exécutée avant l'instruction GOSUB
20	RW	**RESUME sans Error** L'instruction RESUME est exécutée sans qu'une erreur se soit produite.
2	SN	**Erreur de syntaxe** Présence d'une erreur de syntaxe
16	ST	**Forme de chaîne trop complexe** Utilisation d'une forme de chaîne trop complexe.
13	TM	**Inadaptation de types** Le type d'une variable est erroné.
18	UF	**Fonction d'utilisateur indéfinie** Appel d'une fonction qui n'a pas été définie.
8	UL	**Numéro de ligne indéfini.** Un numéro de ligne est spécifié de manière incorrecte.
—	UE	**Erreur indéfinie** Il se produit une erreur dont le code n'est pas spécifié.

Canon

X-07

ORDINATEUR INDIVIDUEL

CARTE DE REFERENCE BASIC